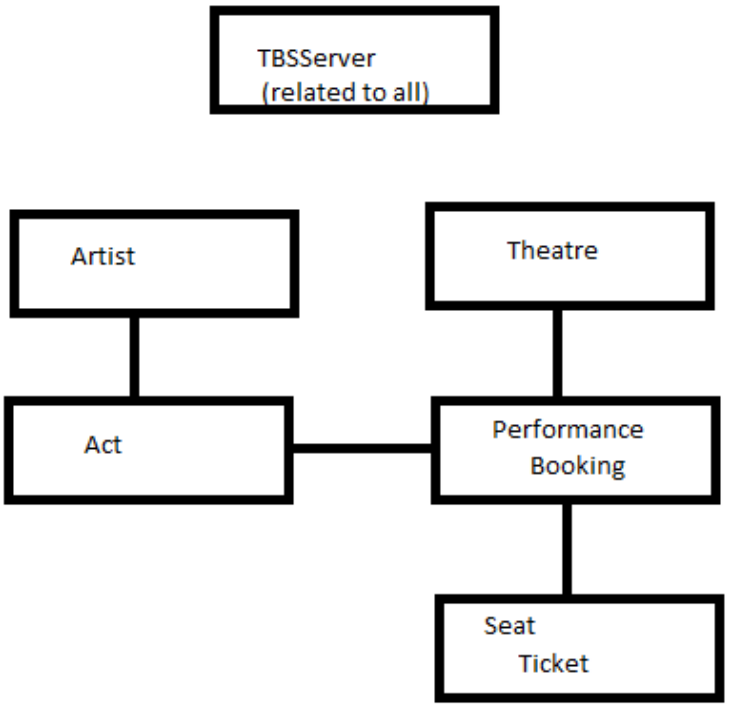
Design of TBSServer: Problem Domain

- Goals of the client is to implement a theatre booking system
- Scenarios
    - Users want to book a seat to their favourite concert
    - User wants to cancel her booking
    - The artist/theatre want to know how much they are making for their act
    - Theatre has limited capacity to host any act (time and space wise) and must be able to schedule any bookings properly
    - User wants to know the price of a seat to a given act
    - Users want to know what is the seating plan
    - Theatre/users/artists want to know how many seats are available
    - Artist wants to know how many acts and when/where they are performing
- Concepts and How are they related?
    - A theatre hosts all the performances
    - Artist plays a number of acts
    - Each act can be performed at different theatres, at different times
    - Each act can be performed more than once at 1 theatre
    - Each performance is of an artists' act
    - A performance has seating arrangement
    - The theatre/performance issues tickets for each seat
    - You make a booking for a seat and get issued a ticket
- Concepts
    - Theatre, artist, act, performance, booking, seat, ticket

Possible relationship between concepts

TBSServer: Implements all core theatre server functions



Artist: Artist contains all acts that the artist perform

Act: Acts can have many performances at different theatres. Acts can have many performances at one theatre at different times

Theatre: Theatre contains details about the theatre itself and all performances it hosts. You can have many performances at a theatre

Performance:

- Performance is of one kind of act
- Performance occurs in a theatre at a given time
- Seats and pricing are allocated for a performance
- Performance has a seating plan
- Tickets can be allocated for a seat to the performance

Seat: Seat has position and availability status within the seating plan. Clients can get information about seat